University of Montenegro
Faculty of Electrical Engineering

Subject: Automated design of electronic circuits and systems

Topic: Ring – Johnson counter

Authors:
Milica Bastrica
Svetlana Zecevic

Date and location:
11.19.2018
Podgorica, Montenegro

Contents

Summary.....1
Description of the problem.....1
Hardware – software solution and simulation.....2
Verification of the work on the board.....7
Video link.....7
Literature used.....7

Summary

Using FPGA board DE2 – 70 we are simulating the work of a Ring – Johnson counter. Switch SW[0] is used for choosing between Ring or Johnson counter. Clock of 2Hz is gained using a clock divider and the fpga board's implemented clock of 50MHz. When the switch SW[0] is off, the work of the Ring counter is simulated. Ring counter is acting as a shift register and after every clock pulsation data from every flip flop is transferred to the next flip flop, and the output of the last flip flop is transferred to the entrance of the first flip flop. When the reset button KEY[0] is pressed first flip flop is set to one, and all the others are set to zero. When the switch SW[0] is on, the work of the Johnson counter is simulated. The Johnson counter works the same way as a Ring counter the only difference is that the output of the last flip flop is inverted and connected to the input of the first flip flop. Once the reset button KEY[0] is pressed , all the flip flops are set to zero.

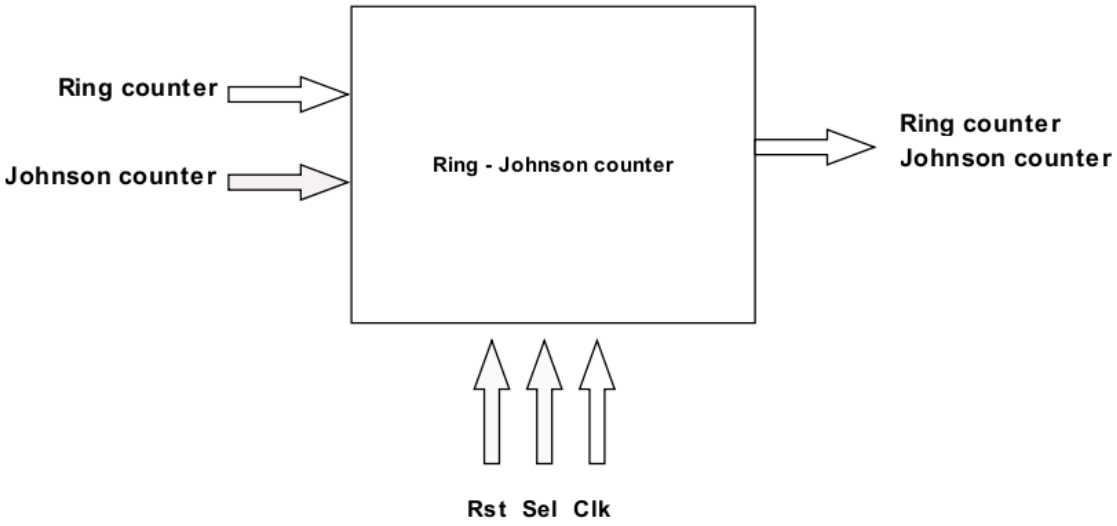
Description of the problem

The assignment is to make a Ring – Johnson counter circuit with the clock of 0.5s. The inputs are reset and clock, and the outputs are four LEDs.

Hardware – software solution and simulation

Altera Quartus 9.1 is used as a software solution. Code is in VHDL and simulations are represented in a WaveForm file.

Simple scheme:



Code:

Divider:

```
entity divider is
generic(N:integer:=25000000);
port
( clk: in std_logic;
clk_new : out std_logic);
end divider;
architecture clk_div_behav of divider is
signal clk_temp : std_logic;
signal temp : integer range 0 to N-1;
        begin
```

```
        process(clk, clk_temp)
        begin
                if(clk'event and clk='0') then
                if(temp=N/2-1)then
                temp<=temp+1;
                clk_temp<='1';
                elsif (temp=N-1) then
                temp <= 0;
                clk_temp<='0';
                else
                temp<=temp+1;
                end if;
        clk_new<=clk_temp;
        end if;
        end process;
end clk_div_behav;
```

Johnson counter:

```
library ieee;
use ieee.std_logic_1164.all;

entity johnsonov_brojac is
port(clk,rst:IN STD_LOGIC;
        output:OUT STD_LOGIC_VECTOR(3 downto 0));
end johnsonov_brojac;

architecture johnsonov_brojac_arh of johnsonov_brojac is
signal temp:STD_LOGIC_VECTOR(3 downto 0):="0000";
```

```
begin
	process(rst,clk)
	begin
		if (rst='0') then
			temp <= "0000";
		elsif (clk'event and clk='1') then
			temp(1) <= temp(0);
			temp(2) <= temp(1);
			temp(3) <= temp(2);
			temp(0) <= not temp(3);
		end if;
	end process;
	output <= temp;
end johnsonov_brojac_arh;
```

Ring counter:

```
library IEEE;
use ieee.std_logic_1164.all;

entity ring_brojac is
port(clk,rst:IN STD_LOGIC;
	output:OUT STD_LOGIC_VECTOR(3 downto 0));
end ring_brojac;

architecture ring_brojac_arh of ring_brojac is
signal temp:STD_LOGIC_VECTOR(3 downto 0):="0000";
begin
	process(clk,rst)
```

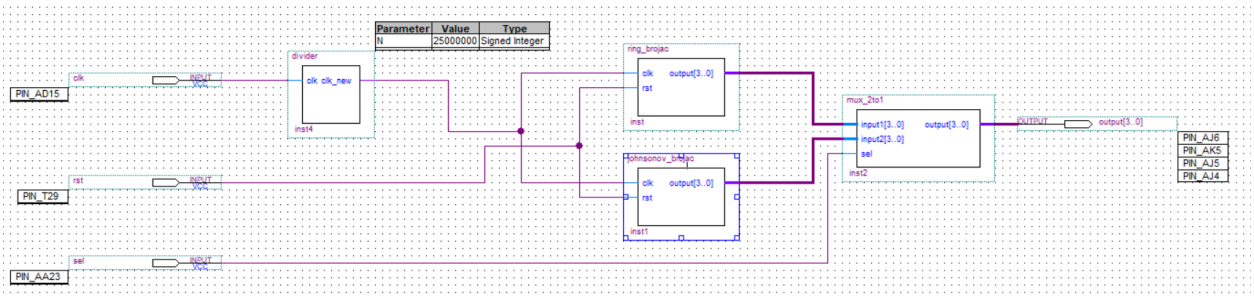
```
	begin
		if (rst='0') then
		temp <= "0001";
		elsif (clk'event and clk='1') then
			temp(1) <= temp(0);
			temp(2) <= temp(1);
			temp(3) <= temp(2);
			temp(0) <= temp(3);
		end if;
	end process;
	output <= temp;
end ring_brojac_arh;
```

Selector:

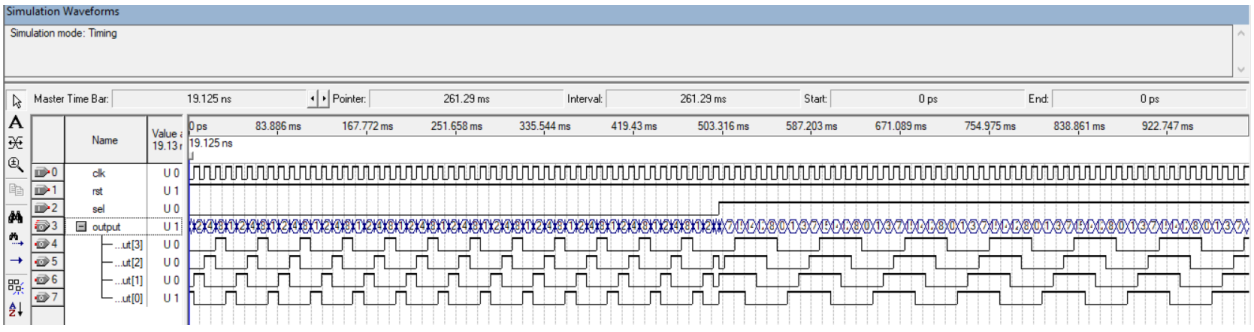
```
library ieee;
use ieee.std_logic_1164.all;
use ieee.std_logic_arith.all;
use ieee.std_logic_unsigned.all;

entity mux_2to1 is
port(input1,input2:IN std_logic_vector(3 downto 0);
sel:IN std_logic;
output:OUT std_logic_vector(3 downto 0));
end mux_2to1;

architecture mux_2to1_arh of mux_2to1 is
begin
	process(sel)
```

```
    begin
        if (sel='0') then
            output <= input1;
        else
            output <= input2;
        end if;
    end process;
end mux_2to1_arh;
```

Bdf file:



Simulation as a wvf file:



FPGA board DE2 – 70 (Cyclone II) is used as a hardware solution

Pinout:

clk PIN_AD15 (50MHz)

output[3] PIN_AJ6 (LED[0])

output[2] PIN_AK5(LED[1])

output[1] PIN_AJ5 (LED[2])

output[0] PIN_AJ4 (LED[3])

rst PIN_T29(KEY[0])

sel PIN_AA23(SW[0])

Verification of the work on the board

Attached link.

Video link

https://youtu.be/d0uzQhBeqiE

Literature used

AUTOMATIZOVANO PROJEKTOVANJE DIGITALNIH SISTEMA (VHDL i FPGA)

DE2_70_User_manual_v109